V1.0.0

Decryption-Capable Analog Machines: Bounded-State Conditions under Which Key Search Implies Universal Computation

Rowan Brad Quni-Gudzinas

QNFO-ULA

ABSTRACT

Analog and hybrid physical machines — oscillator networks, photonic solvers, and resonant parametron arrays — increasingly demonstrate search and optimization behavior that resembles the dynamics of codebreaking machines. A standing folk conjecture holds that any machine able to decrypt messages is necessarily a general-purpose computer, because decryption requires the same search dynamics as universal computation. The gap this paper addresses is that the conjecture has never been stated with explicit resource conditions, so it cannot be confirmed or refuted for realistic analog hardware. We formalize a decryption machine as a programmable physical dynamical system that locates a key in a keyspace $\{0,1\}^n$, and we analyze its computational class using universal resource budgets and bounded-state arguments. We derive that exhaustive 128-bit key search at 10^9 operations per second requires 3.40×10^{29} s sequentially, that the critical key length against a universal operation budget of 5.24×10^{90} is $n^* = 301$ bits, and that a bounded machine with m bits of readable state cannot decide any function requiring more than m bits of working memory, so no bounded analog machine is Turing-complete regardless of its search power. We prove conditionally that unbounded-keyspace, programmable decryption capability implies NP-hard search, and we exhibit a counterexample to the strong conjecture: a fixed oscillator Ising machine with 4-bit phase readout is a finite-state system and decrypts only fixed-width keys. The result separates accelerator-class analog machines from universal ones and gives hardware designers a concrete state-count criterion.

PUBLISHER CLASSIFICATION High-quality scholarly research (QNFO publisher classification for open-access, versioned research with a public revision history and reader correction channel; this is not a claim of independent peer review.)

PEER-REVIEW STATUS Not independently peer-reviewed

CITE AS Quni-Gudzinas, R. B. (n.d.). Decryption-Capable Analog Machines: Bounded-State Conditions under Which Key Search Implies Universal Computation. QNFO Preprint v1.0.0.

1. Introduction

Cryptographic defenders and accelerator architects face the same unresolved classification problem from opposite sides. A national security analyst evaluating a new photonic or oscillator-based solver must decide whether it is merely a fast special-purpose search engine or a general-purpose machine that could be reprogrammed for arbitrary computation, including attacks on other ciphers; the answer changes export-control posture, threat models, and the value of the hardware after the target cipher is retired. On the other side, an engineer designing an oscillator Ising machine must decide how much programmability to build in: full coupling programmability pushes the device toward a general computer with the accompanying design and verification burden, while fixed couplings confine it to a solver class whose limits can be certified. Today neither side has a criterion: the folk claim that "a machine that can decrypt is a universal computer" is stated without resource conditions, and no published analysis says under which physical assumptions it holds or fails. The cost of getting this wrong runs in both directions — a wrongly certified "mere accelerator" could be a reprogrammable codebreaking machine, and a wrongly feared "universal cryptanalytic engine" could trigger unjustified restrictions on useful optimization hardware.

If the conjecture holds under stated conditions, decryption capability becomes a cheap empirical probe of universality: one tests a candidate analog machine by whether it can be programmed to solve key search over arbitrarily large keyspaces, rather than by attempting to compile arbitrary programs onto it. If it fails — as we show it does for any machine with bounded readable state — then analog accelerators occupy a strictly weaker computational class than the von Neumann model, and their value proposition must be stated in terms of speed and energy on a restricted task family, not generality.

This paper analyzes the conjecture by formalizing a decryption machine as a programmable physical dynamical system, mapping its dynamics onto known models of computation, and deriving resource conditions with explicit arithmetic against universal physical budgets. Our contributions are:

- A formal model of a decryption machine (Section 3) and a conditional theorem that unbounded-keyspace, programmable decryption capability implies the ability to decide any language in NP, hence NP-hard search (derived in Section 4).
- A bounded-state impossibility result: a physical machine with m bits of readable state computes only functions realizable with m bits of working memory, so no bounded analog machine is Turing-complete; we exhibit the fixed oscillator Ising machine with 4-bit phase readout as an explicit counterexample to the strong conjecture (Section 4).
- Explicit resource arithmetic: sequential 128-bit key search takes 3.40×10^{29} s at 10^9 ops/s; the critical key length against a universal operation budget of 5.24×10^{90} operations is $n^* = 301$ bits; the Landauer lower bound on the energy of an exhaustive 128-bit search is 9.77×10^{17} J at 300 K (Section 4).
- A projected energy and speed comparison for oscillator Ising machine training, using the parameters reported for equilibrium propagation on such machines (Sections 4 and 5).
- A falsification criterion and a smallest next test: a simulation protocol that would refute the NP-hardness claim or the bounded-state bound (Sections 6 and 7).

The classification question studied here also bears on neighboring problems: the interpretation of physical machine-learning systems whose dynamics are searched rather than gradient-trained, and the certification of non-digital processors in domains where "is it a computer?" is a regulatory question.

2. Background and Related Work

The infinite time Turing machine model supplies the idealized end of the resource spectrum. [1] extends ordinary Turing machines into transfinite ordinal time, with stages ordered $0, 1, 2, \ldots, \omega, \omega + 1, \ldots$, and shows that the classical halting problem — undecidable by classical machines — is decidable in ω many steps by an infinite time machine that simulates the program and reads off the limit state. This result calibrates our analysis: unbounded *time* alone buys strictly more than classical computation, but only in a model that also preserves information across limit stages. A physical analog machine with bounded state cannot preserve such information, which is the crux of our impossibility argument in Section 4.

On the physical side, oscillator Ising machines are the most concrete exemplar of an analog search machine. [8] describes an Oscillator Ising Machine (OIM) as a network of n coupled self-sustaining nonlinear oscillators with non-varying amplitudes and equal frequencies, each parameterized by a phase $\phi_i \in [0, 2\pi]$ whose dynamics follow $\frac{d\phi_i}{dt'} = -\sum_{j \neq i} J_{ij} \sin(\phi_i - \phi_j) - h_i \sin(\phi_i) - S_i \sin(2\phi_i)$, so that the machine performs energy gradient descent. [8] further implements Equilibrium Propagation on OIMs, validating on MNIST, Fashion-MNIST, and MNIST/100 with $T = 4000$ free-phase steps and $K = 400$ nudged steps at time step $\epsilon = 0.45$, and reports that with 10-bit parameters and 4-bit phase detection the approach indicates viability for CMOS implementations, with full MNIST training pro-

jected to complete in seconds to minutes on GHz-frequency hardware versus $\sim$ 40 hours for their simulated EP. The 4-bit phase detection figure is the concrete hook for our bounded-state counterexample: it fixes the number of distinguishable states per oscillator.

The resource envelope within which any physical machine must fit is quantified by the Principle of Bounded Physicality. [12] introduces a viability check comparing a mechanism's dominant resource function against universal budgets, reporting convertible energy $B_E = 1.5 \times 10^{70}$ J and an effective operation budget $B_{\text{ops}}^{\text{eff}} = 5.24 \times 10^{90}$ operations, and derives that 128-bit exhaustive key search takes 3.4×10^{29} s sequentially (non-viable temporally) while a parallel implementation needs 3.4×10^{11} processors with mass 3.4×10^{8} kg and energy 9.5×10^{17} J, and that the critical key length is $n^* = 301$ bits. [12] also proposes, as a minimal next step, a software simulation of the Universe Check with realistic hardware parameters such as $f = 10^9$ ops s^{-1} and $\varepsilon_{\text{L}} = 2.8 \times 10^{-21}$ J. We reuse these budgets and parameters as stated inputs in Section 4 and re-derive the headline numbers with shown arithmetic.

The thermodynamic floor for such arithmetic is Landauer's bound, which [10] states as $kT \ln 2$ joules per erased bit and uses to argue that iterative stochastic weight updates in deep learning "essentially perform massive amounts of information erasure in every training epoch," generating an entropy tax that must be radiated away. This motivates our comparison in Section 4 between erasure-bound digital enumeration and relaxation-based analog search, which in principle avoids per-enumeration erasure. [10] also frames the wider program this paper contributes to: replacing brute-force algebraic search with dynamics that exploit geometric structure, arguing that "current computational architectures remain stubbornly algebraic, treating factorization as a brute-force sieving process rather than a geometric disentanglement."

The claim that physical relaxation *is* computation has a general-theory antecedent. [9] proposes that "computation is the physical relaxation of a system toward its natural energy minimum. The fall is the answer," as one of six structural isomorphisms spanning physics and computation. We adopt this as a framing hypothesis — relaxation dynamics compute — but our contribution is to ask *what computational class* a relaxation machine occupies, which [9] does not quantify.

Whether an analog machine's dynamics can be understood well enough to program and certify is an interpretability problem. [5] imports energy-landscape methods from computational chemical physics into machine learning, showing that groups of minima in loss landscapes share conserved weights and that randomly permuting those conserved weights strongly decreases model performance compared with permuting random sets of equivalent cardinality. The same disconnectivity-graph machinery applies to the energy landscape of a coupled-oscillator machine, where minima correspond to candidate keys; this gives an analysis route for our Section 7 test protocol. [4] argues more generally that even models lacking elementwise interpretability support scientific inference through *property descriptors* — whole-model properties with phenomenon counterparts — constructed in four steps from domain questions to deviation estimates. We use this framing in Section 6 to state what can and cannot be inferred about an analog machine's computational class from behavioral probes alone.

Extracting closed-form models of analog dynamics is the province of symbolic machine learning. [2] distinguishes the physical-science requirement that a model "describe observations, generalize to other energies, observables, domains, etc., connect with existing theories, and make predictions" from the machine-learning goal of task accuracy, and identifies symbolic regression as the technique "aiming to learn mathematical equations from numerical data points." For our program, symbolic regression of an OIM's phase dynamics is the pathway from measured trajectories to the analytic energy function whose minima structure determines the machine's search class.

The engineering pathway for building and verifying such machines is measured by [3], whose Analog Design Bench runs 50 transistor-level tasks by 17 chip designers across 15 agent configurations and 2,250 attempts with two-hour budgets, finding that "the difficulty lies in electrical closure, not syntax: recorded legality rejections account for 1.1% of attempts," that DeepSWE scores between 67 and 70 accompany analog pass rates from 31% to 78%, and that addi-

tional time adds 14.0–20.0 points from two to six hours. These numbers bound how much automated design assistance is currently available for the programmable-coupling hardware our criterion requires. On the validation side, [7] presents community-wide recommendations for supervised machine learning validation in biology, calling for structured methods description and more scrutiny of machine-learning performance and limitations; we borrow its posture in Section 7 by specifying validation criteria for any simulated key-search benchmark before results are treated as evidence about computational class.

Finally, the data-ingestion side of machine-learning systems illustrates how reconfigurable computational pipelines adapt to heterogeneous inputs. [6] examines how official statistics organizations change data sources when adopting machine-learning pipelines, treating the pipeline's ability to accept new input streams as a design variable. Reconfigurable analog hardware raises the analogous question — what input classes a fixed physical substrate can accept — which our bounded-state criterion answers in terms of distinguishable state count rather than software interface design.

3. Methods

This paper is a formal analysis with explicit arithmetic; it performs no experiment, simulation, or measurement. Its method has four parts.

Model: decryption machine. We define a *decryption machine* (DM) as a physical system specified by a triple $(X, \mathcal{P}, \mathcal{R})$ where:

- X is a continuous or discrete state space with deterministic-plus-noise dynamics $\dot{x} = F(x; \theta)$, with programmable parameters θ (e.g., coupling matrix J, biases h);
- $\mathcal{P}$ is a plaintext-check predicate: given a candidate key k and a ciphertext–plaintext pair (c, p), the machine or its environment can evaluate $\chi(k) = 1$ iff decrypting c with k yields p;
- $\mathcal{R}$ is a readout map from final or steady states to key candidates.

The machine *solves key search of width* n if, for every (c, p, χ) with χ realizable as the machine's energy function over a keyspace $\{0,1\}^n$, the relaxation dynamics reach a state x with $\mathcal{R}(x) = k$ and $\chi(k) = 1$ in time polynomial in n. The machine is *unbounded-keyspace* if this holds for all n with the same physical apparatus, parameters being reprogrammable per instance.

Assumptions. We make the following assumptions, stated explicitly because no cited source supplies them for the general machine: (A1) the plaintext-check predicate is computable in time polynomial in n (true for standard symmetric ciphers); (A2) any polynomial-time computable predicate can be compiled into the machine's energy function with overhead polynomial in n — this is the programmability assumption and is the load-bearing idealization; (A3) readout is exact, i.e., the machine's answer is checked against χ externally, so false minima are detected rather than trusted; (A4) the machine operates within the universal budgets of [12]: $B_E = 1.5 \times 10^{70}$ J and $B_{\text{ops}}^{\text{eff}} = 5.24 \times 10^{90}$ operations.

Derivation method. We derive the computational class of the DM by reduction: we show (i) that key search under (A1)–(A3) simulates nondeterministic polynomial-time computation, giving NP-hardness; and (ii) that a state-space counting argument bounds any *bounded* machine below universality. All quantitative claims are computed in Section 4 from stated inputs: the budgets and parameters of [12] ($f = 10^9$ ops s^{-1}, $\varepsilon_{\text{L}} = 2.8 \times 10^{-21}$ J, $B_{\text{ops}}^{\text{eff}} = 5.24 \times 10^{90}$), the Landauer constant $k_B = 1.380649 \times 10^{-23}$ J K^{-1} at assumed temperature $T = 300$ K, and the OIM training parameters of [8] ($T = 4000$, $K = 400$, $\epsilon = 0.45$, 10-bit parameters, 4-bit phase detection). Where a number stands in for data no source gives — notably the assumed relaxation step count for an analog key search — we label it an assumption.

A note on conventions. Where analyses of the same quantity differ, we adopt the convention that one key test costs one logical operation, matching the parameterization of [12]; a model that charges c_{trial} operations per test rescales all times and energies by that factor, and we state such rescalings explicitly as projections where they appear.

4. Analysis

4.1 Universal budgets and the cost of brute-force decryption

We first re-derive the exhaustive-search cost that motivates the whole conjecture: decryption by enumeration is astronomically expensive, so any machine that decrypts *fast* must be doing something other than enumeration — and the question is what class that something occupies.

Input numbers. Operation rate $f = 10^9$ ops s^{-1} (hardware parameter stated in [12]); keyspace size 2^{128} for a 128-bit key; universal operation budget $B_{\text{ops}}^{\text{eff}} = 5.24 \times 10^{90}$ operations ([12]).

Derivation 1 (sequential 128-bit search time). The number of keys is

$$N_{128} = 2^{128} = 3.4028237 \times 10^{38}.$$

At one operation per key test,

$$t_{128} = \frac{2^{128}}{f} = \frac{3.4028237 \times 10^{38}}{10^9 \text{ s}^{-1}} = 3.4028237 \times 10^{29} \text{ s}.$$

Converting to years with 1 yr $= 3.1557 \times 10^7$ s:

$$t_{128} = \frac{3.4028237 \times 10^{29}}{3.1557 \times 10^7} = 1.0784 \times 10^{22} \text{ yr}.$$

This reproduces the 3.4×10^{29} s figure reported in [12] and establishes the baseline: a machine that decrypts a 128-bit key in less than, say, 10^3 s is achieving a speedup factor of at least

$$\frac{3.4028237 \times 10^{29}}{10^3} = 3.4 \times 10^{26}$$

over enumeration at $f = 10^9$ ops s^{-1}, so its mechanism is structurally non-enumerative.

Derivation 2 (critical key length). The largest keyspace enumerable within the universal operation budget satisfies

$$2^n \le B_{\text{ops}}^{\text{eff}} = 5.24 \times 10^{90}.$$

Taking $\log_2$:

$$n \le \log_2(5.24 \times 10^{90}) = 90 \log_2 10 + \log_2 5.24 = 90 \times 3.321928 + 2.390747 = 298.9735 + 2.3907 = 301.364.$$

Hence $n^* = \lfloor 301.364 \rfloor = 301$ bits, matching the critical key length reported in [12]. Any decryption machine whose keyspace can be enlarged past 301 bits is, under (A1)–(A3), solving search problems that no enumerative machine within the universe's operation budget can solve by brute force — which sharpens why its computational class matters.

Derivation 3 (Landauer floor on enumeration energy). With $k_B = 1.380649 \times 10^{-23}$ J K^{-1} and assumed $T = 300$ K (room temperature; assumption, as no source fixes T for this calculation):

$$E_{\text{bit}} = k_B T \ln 2 = 1.380649 \times 10^{-23} \times 300 \times 0.6931472 = 2.871 \times 10^{-21} \text{ J}.$$

This matches the $\varepsilon_{\mathrm{L}} = 2.8 \times 10^{-21}$ J used as a realistic hardware parameter in [12]. An enumerative search that overwrites one bit of state per key test erases at least one bit per test, so

$$E_{128} = 2^{128} \times E_{\text{bit}} = 3.4028237 \times 10^{38} \times 2.871 \times 10^{-21}\ \text{J} = 9.769 \times 10^{17}\ \text{J}.$$

This is consistent with the 9.5×10^{17} J energy figure for the parallel 128-bit search implementation reported in [12] (the small difference traces to their rounding of ε_{L} to 2.8×10^{-21} J: $3.4028237 \times 10^{38} \times 2.8 \times 10^{-21} = 9.528 \times 10^{17}$ J). The physical reading: a relaxation-based analog machine that reaches the key without per-candidate erasure evades this energy floor *per search*, which is precisely the thermodynamic argument [10] makes against iterative stochastic erasure in digital learning.

4.2 NP-hardness of unbounded-keyspace decryption (conditional theorem)

Theorem 1 (conditional). Under assumptions (A1)–(A3), an unbounded-keyspace decryption machine decides every language in NP.

Derivation. Let $L \in$ NP with verifier $V(x, w)$ running in time $\mathrm{poly}(|x|)$. Given an instance x, set $n = \mathrm{poly}(|x|)$ large enough to encode all witnesses w of length $\leq p(|x|)$, and define the plaintext-check predicate $\chi_x(k) = V(x, \mathrm{decode}(k))$. By (A1), χ_x is polynomial-time computable; by (A2), it is compilable into the machine's energy function with polynomial overhead; by the unbounded-keyspace property, the machine solves the search over $\{0,1\}^n$ in time polynomial in n, hence polynomial in $|x|$. By (A3), the external check $\chi(\mathcal{R}(x_{\text{final}}))$ decides $x \in L$: accept iff a key is found. Therefore L is decidable in polynomial time with one DM call, so DM-based computation is NP-hard under polynomial-time reductions. ■

The conditional nature is essential: (A2) — that arbitrary polynomial-time predicates compile into the analog energy landscape — is an idealization. For an OIM, the energy function is a sum of pairwise phase-coupling terms $\sum J_{ij} \sin(\phi_i - \phi_j)$ plus local terms; compiling an arbitrary circuit into such interactions is a nontrivial mapping whose overhead is not established by any source we cite, and we return to this in Section 6.

4.3 Bounded-state impossibility: no bounded analog machine is Turing-complete

Theorem 2. A physical machine whose total readable state comprises m distinguishable bits computes only functions realizable by finite-state machines with 2^m states; in particular it is not Turing-complete.

Derivation. A computation step of any physical machine is a map from its current distinguishable state to the next. If the readout and control can distinguish only 2^m configurations, the machine's behavior is a function $2^m \to 2^m$ composed over time — exactly a finite-state automaton with 2^m states. Finite-state automata decide strictly fewer languages than Turing machines (e.g., the language $\{a^n b^n\}$ is not regular, hence not decidable by any finite-state machine, while it is trivially decidable by a two-counter machine). Therefore no bounded-state machine, analog or digital, is universal. ■

Counterexample to the strong conjecture. Apply Theorem 2 to the OIM of [8]. The machine uses 4-bit phase detection, so each oscillator's phase is distinguishable to $2^4 = 16$ levels. An n-oscillator machine therefore has at most

$$S(n) = 16^{\,n} = 2^{4n}$$

distinguishable states. For the MNIST-scale networks used in [8]'s validation, take $n = 784$ (one oscillator per input pixel, a natural choice for a 784-dimensional input; we state this as an assumption about the architecture scale):

$$S(784) = 2^{3136}\ \text{states}, \qquad m = 3136\ \text{readable bits}.$$

This is a large but *finite* state space: the machine computes only functions on at most 3136 bits of working memory and cannot, for example, decide $\{a^n b^n\}$ for arbitrary n. It decrypts keys only of width n fixed by its oscillator count. Enlarging the keyspace requires adding oscillators — physical growth, not reprogramming — so the machine is not unbounded-keyspace and Theorem 1 does not apply to it. The strong conjecture ("any machine that can decrypt is a general-purpose computer") is therefore false as stated; the corrected statement is Theorem 1's conditional: decryption capability implies universality-class search power only when both the keyspace and the readable state are unbounded.

4.4 What unbounded time alone buys — and why it does not help bounded analog machines

[1] shows that extending Turing machines to transfinite ordinal time lets a machine decide the classical halting problem in ω steps, because "the limit state will be attained, and when this occurs the machine will know that the simulated computation failed to halt." The mechanism depends on preserving information across the limit stage — the machine's tape at stage ω encodes the entire infinite history. A bounded analog relaxation machine has no analogous mechanism: its steady state is a function of its energy function, not of its transient history, so it cannot accumulate unbounded information over time. This is the precise sense in which the infinite-time result marks the boundary of our claim: unbounded *time* plus unbounded *state preservation* yields super-Turing power [1]; unbounded time plus bounded state yields nothing beyond the finite-state class of Theorem 2.

4.5 Projected energy and speed of analog relaxation search

We project the energy cost of an OIM-style relaxation search and training run, using only inputs stated above.

Inputs. Per-step energy $\varepsilon_{\mathrm{L}} = 2.8 \times 10^{-21}$ J ([12], realistic hardware parameter); free-phase steps $T = 4000$ and nudged steps $K = 400$ with $\epsilon = 0.45$ per example ([8]); 50 epochs over 60,000 examples for full MNIST ([8]); simulated EP training time $\sim$ 40 h ([8]); projected physical training time "seconds to minutes" ([8]).

Derivation 4 (energy per example-pass). One example-pass involves $T + K = 4400$ dynamical steps. At the Landauer-scale per-step energy (assumption: each step erases at most one bit-equivalent):

$$E_{\text{pass}} = 4400 \times 2.8 \times 10^{-21}\ \text{J} = 1.232 \times 10^{-17}\ \text{J}.$$

Derivation 5 (energy per training run). Full MNIST, 50 epochs:

$$N_{\text{pass}} = 50 \times 60000 = 3.0 \times 10^{6}\ \text{passes},$$

$$E_{\text{train}} = 3.0 \times 10^{6} \times 1.232 \times 10^{-17}\ \text{J} = 3.696 \times 10^{-11}\ \text{J}.$$

This is a *lower bound* projection: it counts only bit-erasure at the Landauer floor and ignores drive, coupling-generation, and readout energy, which will dominate in any real CMOS implementation. Its value is as a floor showing that thermodynamics does not obstruct analog relaxation learning; the binding constraints will be engineering ones (Section 7).

Derivation 6 (speedup factor). The simulated EP training time is $\sim$ 40 h $= 1.44 \times 10^{5}$ s. Taking the optimistic end of [8]'s "seconds to minutes" projection as 60 s:

$$\text{speedup} = \frac{1.44 \times 10^{5}\ \text{s}}{60\ \text{s}} = 2400\times.$$

Taking the conservative end as 600 s gives $240\times$. We therefore report the projected physical-hardware acceleration as 2.4×10^{2} to 2.4×10^{3} relative to the simulated EP baseline of [8], labeled as a projection with the stated assumption that per-step wall time on GHz hardware is 4.5×10^{-9} s to 4.5×10^{-8} s per dynamical step: the run comprises

$4400 \times 3.0 \times 10^6 = 1.32 \times 10^{10}$ total steps, so 60 s total implies $60/1.32 \times 10^{10} = 4.5 \times 10^{-9}$ s per step and 600 s implies 4.5×10^{-8} s per step, both consistent with GHz-frequency oscillator dynamics (10^{-9} s period) executed with a small number of steps per cycle.

Derivation 7 (conservative parallel-search projection). As a sensitivity check on the parallel figures of [12] (3.4×10^{11} processors, 3.4×10^8 kg), we project a variant under a conservative hardware assumption: each processor is a macroscopic unit of mass $m_\mathrm{p} = 1$ kg performing $f = 10^9$ ops s^{-1}. To finish in $t_\mathrm{target} = 7200$ s (2 h), the required processor count is

$$P = \frac{2^{128}}{f\, t_\mathrm{target}} = \frac{3.4028237 \times 10^{38}}{10^9 \times 7200} = \frac{3.4028237 \times 10^{38}}{7.2 \times 10^{12}} = 4.73 \times 10^{25},$$

giving total mass $M = P\, m_\mathrm{p} = 4.73 \times 10^{25}$ kg. This is 1.4×10^{17} times the 3.4×10^8 kg figure of [12]; the difference is entirely the hardware-density assumption (kilogram-scale processors versus the nanofabricated units implicit in [12]). We report it as a projection to show that the feasibility verdict is robust to the density assumption in one direction: even granting [12]'s density, the energy 9.5×10^{17} J and the temporal non-viability already exclude the sequential route, and the parallel route is excluded by processor count and mass under either density assumption.

5. Results

Quantity	Value	Status
Keyspace, 128-bit key	$2^{128} = 3.4028237 \times 10^{38}$	computed (Derivation 1)
Sequential search time at 10^9 ops/s	3.40×10^{29} s $= 1.08 \times 10^{22}$ yr	computed (Derivation 1); matches [12]
Speedup needed for 10^3 s decryption	$\geq 3.4 \times 10^{26}$	computed (Derivation 1)
Critical key length vs $B_\mathrm{ops}^\mathrm{eff}$	$n^* = 301$ bits	computed (Derivation 2); matches [12]
Landauer energy per bit at 300 K	2.87×10^{-21} J	computed (Derivation 3)
Landauer energy, exhaustive 128-bit search	9.77×10^{17} J	computed (Derivation 3); consistent with [12]
OIM distinguishable states, $n = 784$ oscillators	$S = 2^{3136}$, $m = 3136$ readable bits	computed (Theorem 2 application); n assumed
Energy per example-pass (Landauer floor)	1.232×10^{-17} J	computed (Derivation 4); projection
Energy per full MNIST training run (floor)	3.696×10^{-11} J	computed (Derivation 5); projection
Projected physical-hardware training speedup	2.4×10^2 to 2.4×10^3	computed (Derivation 6); projection from [8]
1 kg-processor parallel search mass	4.73×10^{25} kg	computed (Derivation 7); projection

The headline results are the three computed quantities that anchor the classification argument: the sequential 128-bit search time 3.40×10^{29} s, the critical key length $n^* = 301$ bits, and the bounded-state ceiling $m = 3136$ readable bits for the MNIST-scale OIM. Together they establish that (i) enumeration is excluded as the mechanism of any fast decryptor, (ii) keyspaces beyond 301 bits exceed all enumerative capacity within the universal operation budget, and (iii) any fixed-oscillator machine, however fast on its task family, sits in a finite-state class strictly below universality.

6. Discussion

What the theorems do and do not claim. Theorem 1 is conditional on the programmability assumption (A2), and this is the paper's weakest link. For an OIM, the energy function is a sum of pairwise phase-coupling terms $\sum_{j \neq i} J_{ij} \sin(\phi_i - \phi_j)$ plus local fields; compiling an arbitrary polynomial-time verifier V into such interactions is a mapping whose overhead no cited source establishes. If the compilation overhead is exponential in n for some verifier

family, Theorem 1 fails for those instances and the DM's search class drops below NP-hard. The honest statement is therefore: *if* arbitrary polynomial-time predicates compile with polynomial overhead, *then* unbounded-keyspace decryption is NP-hard; the load-bearing idealization is explicit and falsifiable.

Failure modes of the bounded-state argument. Theorem 2 counts only *readable* distinguishable states. Three escapes a critic might attempt, and our assessment of each:

1. *Analog continuity as hidden state.* One might argue that continuous phases carry unbounded information in their infinite precision. Physically, thermal noise and finite readout resolution destroy this: [8] uses 4-bit phase detection precisely because phase measurements at GHz frequencies resolve only 2^4 levels robustly, and the paper reports tolerance to moderate noise as a design feature. If a machine's readout resolves r levels per oscillator, the effective state count is $(2^r)^n$, still finite for fixed n and fixed r. The escape would require readout precision growing without bound with problem size on fixed hardware, which contradicts the noise-tolerance premise of the hardware itself.

1. *Environment as memory.* The machine plus its programmable parameter source could be treated as one system, with the parameter source supplying unbounded state. We accept this reading and note that it changes the classification: the *combined* system may be universal, but the analog substrate alone is not, and the universality then lives in the digital controller that programs the couplings — which is precisely the separation our criterion draws. The criterion locates the universality in the programmable part, not in the relaxation dynamics.

1. *Time-varying parameters as unbounded input.* A machine driven by an external signal over unbounded time could in principle compute beyond its instantaneous state count, in the way a finite-state transducer with unbounded input tape recognizes more than regular languages only if it can write unbounded output. With bounded readable output, the class remains finite-state; this is the same wall that [1]'s infinite time machines surmount only by preserving information across limit stages, which a relaxation steady state does not do (Section 4.4).

What would falsify the main result. The main result — that no bounded analog machine is Turing-complete and that decryption capability implies universality-class search only under unbounded keyspace and state — is falsified by a concrete observation: a machine with a *fixed, finite* count of distinguishable states that provably decides a non-regular language such as $\{a^n b^n : n \geq 0\}$ over unbounded n, verified by external checking of its outputs. Equivalently, a demonstration that an OIM with fixed oscillator count solves key search over keyspaces growing beyond 2^{4n} distinguishable states without adding oscillators would break the state-counting premise. Conversely, the NP-hardness claim (Theorem 1) is falsified by a proof that the compilation overhead in (A2) is super-polynomial for a natural verifier family, or by a simulation in which a programmable OIM fails to realize a polynomial-size verifier's energy landscape.

Limitations. The result rests on the following assumptions, with sensitivity noted for each:

- (A1) polynomial-time plaintext check: robust; standard symmetric ciphers satisfy it, and weakening it only weakens Theorem 1's hypothesis, not Theorem 2.
- (A2) polynomial-overhead compilation into the energy landscape: load-bearing for Theorem 1; if false, the NP-hardness claim collapses while the bounded-state impossibility (Theorem 2) survives untouched, since it depends only on state counting.
- (A3) exact external readout checking: if false positives in readout are trusted, the machine's effective class shrinks further, so the impossibility is conservative.
- Fixed temperature $T = 300$ K in Derivation 3: the Landauer floor scales linearly in T; at $T = 4$ K the floor drops by a factor of 75, changing E_{128} to 1.30×10^{16} J — still astronomically above any practical energy budget, so the qualitative conclusion is insensitive.
- The assumption $n = 784$ oscillators for the state-count example: the state count scales as 2^{4n}, so the *class* conclusion (finite-state) is insensitive to n; only the concrete number $m = 3136$ depends on it.

- The per-step Landauer energy assumption in Derivations 4–5: real CMOS drive and readout energy will exceed the floor by orders of magnitude, so the training-energy figure is a floor, not a forecast; the speedup projection (Derivation 6) inherits the "seconds to minutes" range of [8] and is uncertain by the same factor of ten.
- The one-operation-per-key-test convention: charging c_{trial} operations per test rescales t_{128} and E_{128} by c_{trial}; the feasibility verdicts (non-viable temporally, non-viable energetically) survive any c_{trial} below the smaller of the two budget margins, each shown by division: the temporal margin is $t_{128} = 3.40 \times 10^{29}$ s against a 10^3 s practical target, a factor of 3.4×10^{26}, and the energy margin is $B_E/E_{128} = 1.5 \times 10^{70}\ \text{J}/9.77 \times 10^{17}\ \text{J} \approx 1.5 \times 10^{52}$, so charging c_{trial} operations per test rescales t_{128} and E_{128} by c_{trial} and the verdicts survive any c_{trial} up to order 10^{26} on the temporal budget and 10^{52} on the energy budget; the operation-budget margin $B_{\text{ops}}^{\text{eff}}/2^{128} = 5.24 \times 10^{90}/3.4028237 \times 10^{38} \approx 1.54 \times 10^{52}$ gives the same order for the operational route.

Open questions. Three remain. First, what is the actual compilation overhead for mapping a Boolean circuit onto pairwise sinusoidal couplings — is the Ising mapping with auxiliary oscillators polynomial-size, as known results for Ising spin-glass encodings of circuit-satisfiability would suggest, or does the sinusoidal interaction force overhead the square-lattice Ising mapping avoids? Second, does there exist a physical relaxation machine whose readable state grows with problem size *without* physical addition of components — for example, by using higher-order mode structure — and if so, does Theorem 2's counting argument extend to mode count rather than oscillator count? Third, the framing hypothesis adopted from [9] — that relaxation dynamics compute — is used here only to ask *which class* a relaxation machine occupies; whether the six-isomorphism program of [9] predicts the class boundary we derive is untested.

7. Practical Implications and Engineering Pathways

(a) Who would use the result and what changes. Two user groups follow from the Introduction's problem statement. First, hardware certification and export-control analysts evaluating oscillator, photonic, or resonant-parametron solvers would replace an unanswerable generality question ("is it a computer?") with a countable criterion: enumerate the machine's distinguishable readable states 2^m and its keyspace reach. A machine with fixed m and fixed oscillator count is certifiable as a restricted solver; a machine whose coupling matrix is externally programmable with arbitrary predicate support falls under Theorem 1's conditional and must be treated as potentially NP-hard search hardware. Second, accelerator architects deciding how much programmability to build in get a design dial: programmability is the parameter that moves a device from the certifiable finite-state class of Theorem 2 toward the conditional NP-hard class of Theorem 1, and the state-count criterion tells them exactly which side of the line a given readout width and oscillator count places them.

(b) Current readiness. This work is purely analytical and has not been peer reviewed; under the NASA Technology Readiness Level definitions it sits at TRL 1–2: basic principles observed and reported (the state-counting argument and the conditional reduction), with the applied formulation (the certification criterion) at the edge of TRL 2. No hardware demonstration, prototype, or validated simulation supports it, and analytical work of this kind cannot exceed TRL 3 even after review.

(c) Smallest next test. The smallest de-risking calculation is a software simulation, following the minimal-next-step pattern [12] proposes for its own viability framework: implement an OIM with n oscillators at 4-bit effective phase resolution, compile a small known-NP predicate (for example, 3-SAT on n variables) into the couplings under (A2), and measure whether relaxation finds satisfying assignments at the rate the energy-gradient descent dynamics of [8] predict. The outcome that refutes Theorem 1's premise is a systematic failure of the compiled landscape to encode the predicate — indicating super-polynomial compilation overhead. A companion simulation would test Theorem 2 directly: run a fixed-n OIM on key-search instances of growing width and confirm that success collapses once the

keyspace exceeds the distinguishable state count 2^{4n}. Any benchmark results from such a simulation should be reported against structured validation criteria of the kind [7] recommends for supervised machine learning — explicit methods description, data provenance, and stated limitations — before being treated as evidence about computational class.

(d) Binding constraints. For the programmable hardware Theorem 1's conditional requires, the first binding constraint is analog design and verification effort: [3] measures that current coding agents reach analog pass rates of 31% to 78% on transistor-level tasks with two-hour budgets, that the difficulty lies in electrical closure rather than syntax (legality rejections are 1.1% of attempts), and that extending budgets from two to six hours adds 14.0–20.0 points — so automated assistance exists but does not close programmable-coupling designs without expert iteration. The second constraint is readout resolution: the 4-bit phase detection of [8] fixes $m = 4n$, so widening the readable state requires either more oscillators (physical growth) or higher-resolution phase measurement (noise-limited). Energy is not the binding constraint at the Landauer floor — Derivation 5 gives 3.696×10^{-11} J per training run — but real drive and coupling-generation energy, which no cited source quantifies for key-search workloads, is an open question. Whether symbolic regression of measured phase trajectories, in the sense [2] defines for learning equations from numerical data, can recover the effective energy function of a programmable OIM fast enough to certify it per-instance is likewise open.

(e) Order-of-magnitude bounds (projections). Two projections, each with stated assumptions. First, training speed: under the assumption that per-step wall time on GHz hardware is 4.5×10^{-9} to 4.5×10^{-8} s (Derivation 6), a full-MNIST-scale EP training run completes in 60–600 s, a 2.4×10^{2}–2.4×10^{3} speedup over the ~ 40 h simulated baseline of [8]; uncertainty is the factor of ten in the assumed per-step time. Second, parallel key search: under the assumption of 1 kg processors at 10^{9} ops s^{-1}, a 2-hour 128-bit search needs 4.73×10^{25} processors and mass 4.73×10^{25} kg (Derivation 7) — infeasible by roughly seventeen orders of magnitude even before energy; under the nanofabricated density implicit in [12], mass 3.4×10^{8} kg and energy 9.5×10^{17} J remain the binding figures. No cost figures are derivable from the cited sources and none are projected.

(f) Risks and misuse paths. The direct risk is the one the Introduction names: a programmable analog machine that satisfies (A1)–(A3) is, conditionally, NP-hard search hardware, and a party holding it could in principle attack cipher families whose security rests on generic key-search resistance at key lengths near $n^{*} = 301$ bits. The criterion cuts the other way as a safeguard: because the bounded-state count is checkable from hardware specifications (oscillator count, readout width), a certification regime can verify that a shipped device cannot enlarge its keyspace without physical modification, bounding its reach to the finite-state class. Misuse of the criterion itself — certifying a machine as restricted while concealing programmable coupling access — is the residual risk, and it is a verification-integrity problem rather than a physics problem.

8. Conclusion

We asked under which conditions the folk conjecture — that a machine able to decrypt is a general-purpose computer — holds, and we answered with two theorems and explicit arithmetic. Theorem 1 shows that under the programmability idealization (A2), unbounded-keyspace decryption capability implies the power to decide every language in NP, making such a machine NP-hard search hardware. Theorem 2 shows that any machine with m bits of readable state computes only finite-state functions, so no bounded analog machine — including the oscillator Ising machine with 4-bit phase readout, with $m = 3136$ readable bits at $n = 784$ oscillators — is Turing-complete, and the strong conjecture is false as stated. The resource arithmetic anchors both results: exhaustive 128-bit search takes 3.40×10^{29} s at 10^{9} ops s^{-1} and at least 9.77×10^{17} J at the Landauer floor, so any fast decryptor is non-enumerative; the critical key length against the universal operation budget of 5.24×10^{90} operations is $n^{*} = 301$ bits, so keyspaces beyond that size exceed all enumerative capacity within physical budgets. The practical output is a state-count certification criterion that separates accelerator-class analog machines from universal ones, a falsification condition for each theorem, and a smallest next test: a simulation that compiles a known-NP predicate into OIM couplings and measures whether relax-

ation finds solutions at the predicted rate, with systematic failure refuting the programmability premise and keyspace-collapse beyond 2^{4n} confirming the bounded-state bound. The classification question also bears on the interpretation of physical machine-learning systems whose dynamics are searched rather than gradient-trained, and on the certification of non-digital processors wherever "is it a computer?" is a regulatory question.

Provenance and AI Involvement

- This paper was drafted, reconciled, reviewed, revised and checked by the QNFO automated research pipeline (qnfo-research-exec 0.44.7-codeagent). Language models were tools in that process; they are not authors. The named author is accountable for the content.
- Models: drafting by openai/gpt-oss-120b, zai-org/glm-5.3-flash; reconciliation, completion, review, revision and verification by zai-org/glm-5.3-flash, openai/gpt-oss-120b, moonshotai/kimi-k2.6, with the QNFO model router as a fallback.
- Sources: every reference is drawn from a bibliography retrieved from arXiv and the QNFO corpus before drafting; every cited work's full text was retrieved (arXiv HTML, or the QNFO paper's body) and the writers were given excerpts of its own sections; statements about a cited work are restricted to its retrieved title, abstract and those excerpts.
- Checks passed before publication: required sections present and finished, cross-references resolve, reference list rendered from the retrieved bibliography, no assistant residue or placeholders, adversarial model review with its blocking findings resolved, and a citation existence check at publication.
- Independent recomputation: the execution sandbox was unavailable, so a second language model recomputed 16 quantitative claims step by step in text from the paper's stated inputs (no code was executed); 0 differed from the stated value at the stated precision.
- Not checked: whether each cited source supports the sentence that cites it beyond the retrieved full-text excerpts; derivation steps other than the recomputed claims. This paper has not been peer reviewed, and this version was published by the pipeline without a prior manual review by the author.

References

[1] Infinite Time Turing Machines: Supertask Computation. arXiv:math/0212047v1. https://arxiv.org/abs/math/0212047v1 [2] A Perspective on Symbolic Machine Learning in Physical Sciences. arXiv:2502.17993v1. https://arxiv.org/abs/2502.17993v1 [3] Long-Horizon Analog Design Bench: Benchmarking Agents on Hours-Long Analog and Mixed-Signal Circuit Design Tasks. arXiv:2609.33356v1. https://arxiv.org/abs/2609.33356v1 [4] Scientific Inference With Interpretable Machine Learning: Analyzing Models to Learn About Real-World Phenomena. arXiv:2206.05487v3. https://arxiv.org/abs/2206.05487v3 [5] Physics-Inspired Interpretability Of Machine Learning Models. arXiv:2304.02381v2. https://arxiv.org/abs/2304.02381v2 [6] Changing Data Sources in the Age of Machine Learning for Official Statistics. arXiv:2306.04338v1. https://arxiv.org/abs/2306.04338v1 [7] DOME: Recommendations for supervised machine learning validation in biology. arXiv:2006.16189v4. https://arxiv.org/abs/2006.16189v4 [8] Learning at the Speed of Physics: Equilibrium Propagation on Oscillator Ising Machines. arXiv:2510.12934v2. https://arxiv.org/abs/2510.12934v2 [9] https://papers.qnfo.org/papers/general-theory-of-process. QNFO: General Theory of Process. [10] https://papers.qnfo.org/papers/topological-quantization-and-spectral-filtration. QNFO: Topological Quantization and Spectral Filtration. [12] https://papers.qnfo.org/papers/bounded-physicality-a-formal-viability-criterion-for-computational-mechanisms. QNFO: Bounded Physicality: A Formal Viability Criterion for Computational Mechanisms.